**Lab 04 Tasks**

**Task1**

Create a Java Swing application with two forms: the first form for user registration and the second to display the registered data in a JTable. The registration form should include fields for Full Name, Father's Name, CNIC, Phone Number, Password, Confirm Password, Gender, Nationality, and Email, along with a Register button. When the Register button is clicked, validate that all fields are filled and that the Password matches the Confirm Password. If any validation fails, display an error message using a JOptionPane. If the registration is successful, open the second form and display the user's data in a JTable.

**Task2**

Create a Java Swing application that features a custom-designed keyboard interface. When a key on this virtual keyboard is pressed, the application should display the key that was pressed along with its corresponding ASCII code. This task will help you understand event handling in Swing, particularly how to capture and process keyboard inputs, and display relevant information dynamically.

**Task3**

Create a Java Swing application that simulates an ATM machine, featuring a login screen where users input their account number and PIN to access the main menu. The main menu should offer options to check the account balance, deposit money, withdraw money, and exit. Ensure that the application validates the login credentials, prevents users from withdrawing more than their available balance, and only allows positive amounts for deposits and withdrawals. Use JOptionPane for error handling, such as when an invalid PIN is entered, insufficient funds are available for a withdrawal, or an invalid amount is input. The application should simulate a typical ATM session with secure login, balance checks, deposits, and withdrawals, and handle errors appropriately.

**Task4**

Develop a Java Swing application to enable students to calculate their SGPA for a semester. The application should include a form for inputting student details such as Name, Roll Number, and Class, along with fields for entering the grades and credit hours for various subjects. Implement a "Calculate SGPA" button to compute and display the SGPA in a non-editable field or label. Provide a "View Summary" button to show a performance summary, including the grades, credit hours, calculated SGPA, and a performance remark based on the SGPA Include a "Reset" button to clear all fields. For example, if a student inputs grades for subjects with corresponding credit hours (e.g., Math: 8 (4 credits), Science: 9 (3 credits), English: 7 (2 credits)), the application should compute the SGPA based on these values and display it along with a summary and remark. The app should also handle invalid or incomplete data entries with alerts.

**Task5**

Objective: Develop a Java Swing application that demonstrates real-time text transformation based on user input. The application should include a text area for inputting text and a label to display transformed text. Implement a KeyListener to monitor key presses in the text area and update the label with the transformed text. For example, apply transformations such as converting the text to uppercase, reversing the text, or replacing spaces with underscores. Provide a "Transform" button that triggers the transformation based on the selected option. Include radio buttons or a dropdown menu for users to choose the type of transformation. For instance, if a user types "hello world" and selects "Reverse," the label should display "dlrow olleh." Ensure the application updates the label in real-time as text is entered, and handle edge cases like empty input gracefully.